The Mars Exploration Rover / Collaborative Information Portal

Joan Walton
NASA
Ames Research Center
Moffett Field, CA 94035
U.S.A.
Joan.D.Walton@nasa.gov

Robert E. Filman
RIACS
Ames Research Center
Moffett Field, CA 94035
U.S.A.
rfilman@arc.nasa.gov

John Schreiner
NASA
Ames Research Center
Moffett Field, CA 94035
U.S.A.
John.A.Schreiner@nasa.gov

Abstract

We describe the architecture and interface of the Mars Exploration Rover/Collaborative Information Portal (MER/CIP), a system for integrating operational and scientific information for managing the 2003 Mars Exploration Rovers. MER/CIP displays schedules, notifies of events and the arrival of scientific data products, displays those products, and facilitates collaboration, all within the context of user-personalized access and interfaces. MER/CIP is an Internet Java application connecting to a web-services–based middleware and data- and meta-database back ends.

1   Managing Mars Rovers

Astrologers argue that the alignment of the planets governs human affairs. Scientists scoff at this idea. However, there is one important exception, when planetary alignment matters to science: launch windows for planetary exploration. In late May and early June, 2003, Mars will be in position. Two Mars Exploration Rovers (MER) (Figure 1) will rocket towards the red planet. The Rovers will perform a series of geological and meteorological experiments, examining geological evidence for water and conditions once favorable for life (Cornell/Athena 2003; Jet Propulsion Laboratory, 2003).

Back on earth, a small army of surface operations staff will work to keep the Rovers running, sending directions for each day's operations and receiving the files encoding the outputs of the Rover's six instruments. (Mars is twenty light minutes from Earth. The Rovers must be fairly autonomous.) The fundamental purpose of the project is, after all, Science. Scientists have experiments they want to run. Ideally, scientists want to be immediately notified when the data products of their experiments have been received, so that they can examine their data and (collaboratively) deduce results and plan the following experiments.
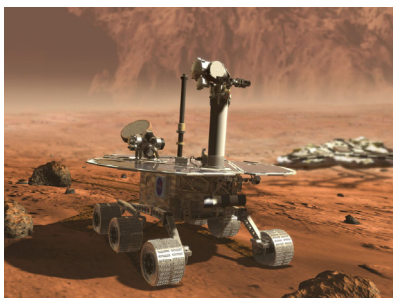

Figure 1: An artist's rendition of a Mars Exploration Rover on Mars

Mars is an unpredictable environment. We issue commands to the Rovers. However, we don't know if the commands will execute successfully, nor which of the objects sensed by the Rovers will be worthy of further examination. The steps of what, to a scientist, are conceptually individual experiments may be intermixed and scattered over a large number of activities. While the scientific staff has an overall strategic idea of what it would like to accomplish, concrete activities are planned daily. The data and surprises of the previous day need to be integrated into the negotiations for the next day's activities, all synchronized to a schedule of

transmission windows. "Negotiate" is the operative term, as different scientists want the same resources to run possibly incompatible experiments. Many meetings plan each day's activities.

The Mars Exploration Rover/Collaborative Information Portal (MER/CIP) provides a centralized, one-stop delivery platform integrating science and engineering data from several distributed heterogeneous data sources. Key issues that MER/CIP addresses include

- *Scheduling and schedule reminders*. Operations planning is driven by meetings. Participants need dynamic information about where they need to be and cross-correlation to activities on Mars. Rather shortsightedly, all extant calendar tools presume a 24 hour day; A Martian Sol is 24.66 hours. Scheduling with respect to Mars time is critical, for the Rover is powered by sunlight. For scheduling, both time-scales must be visible.
- *Tracking the status of daily predicted outputs*. The outputs of experiments are radio-transmitted to earth daily. Experimenters want know what is planned to be done on a given Sol and what actually happened. They need to be informed when their data products have arrived. However, it is difficult to track the path from scientific command through the interleaving of command execution on to the data.
- *Finding and analyzing data products*. This includes searching through the data-products space and analyzing the data files found there. Such examinations can be as simple as viewing pictures or as complex as running scientist-created data analysis software. The data is stored in existing heterogeneous structures, developed independently and obliviously to the needs of the portal.
- *Collaboration*. Scientists and operations managers want to share information including not only data products, but also the results of analyses and annotations of these products and analyses.
- *Announcements*. MER/CIP serves as a primary mechanism for broadcasting to the staff announcements of events such as changes in schedules of meetings.
- *Personalization*. User interfaces, data product awareness and access rights to data must be personalized to the preferences and rights of each user. In particular, MER/CIP servers two very different communities: scientists, interested in the results of particular experiments, and operations staff, responsible for maintaining Rover health and safety.

2 Goals

Our goal in developing MER/CIP (and related projects, an emerging technology we call the *Info-Core Information Infrastructure*) (Walton, Filman, Knight, Korsmeyer & Lee, 2001; Walton, Filman, & Korsmeyer 2000) has been to create a generic information infrastructure for integrating scientific and engineering data. Key elements of this domain include:

- Integrating heterogeneous data sources
- Managing large amounts of data
- Supporting the use of unstructured data
- Controlling access to data in a distributed and possibly federated environment according to the rights and privileges of particular users
- Facilitating collaboration
- Providing tools for browsing and analyzing a range of data
- Presenting quality interfaces for the above tasks
- Doing all this in a familiar, easily installed and manipulated environment.

Figure 2 shows a screen-shot from the current version of MER/CIP. Key elements of this interface include the simultaneous access to a variety of different information sources using a variety of
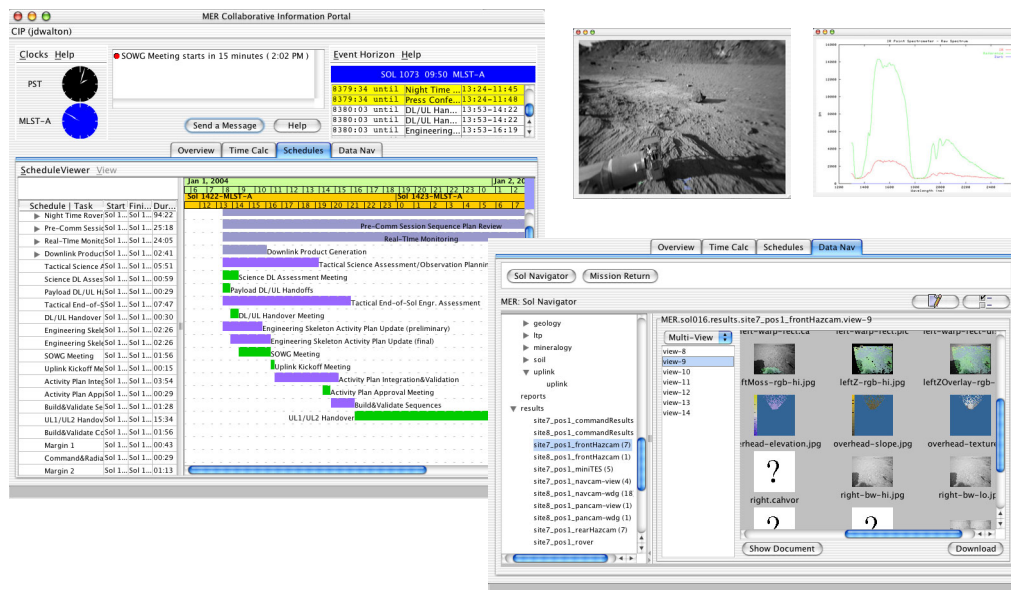
Figure 2: The MER/CIP user interface

GUI themes; the integration of numeric, structured, and photography information; scheduling tools based on Mars time, and the implementation of the system as a Java application.

3 MER/CIP Architecture

Architecturally, MER/CIP is a client-server web-services Java application (Figure 3). Key elements of this architecture are

- *Clients are applications.* The client program is a Java application, delivered over the net as a self-installing Jar file. The client application accesses services through web services and Java messaging. "Clients are applications" contrasts with our original design of "clients are applets in web browsers." Developers have more control over the organization and quality of interaction of applications than applets. We can skip debugging in each version of each browser. A Java application as a net-accessible program is enabled by recent implementations of executable jar files. However, running as an application requires us to manage our own security and encryption, and limits the variety of data types that are easily rendered. Java classes that render HTML are common; Java classes that substitute for exotic browser plug-ins are not. Fortunately, our application does not need the latter. In practice, the transition from applets to application has proven straightforward.

- *Tabbed pane GUI.* The dominant theme of the client GUI is a collection of subapplications on tabbed panes beneath a "management area." The management area contains tools for quickly viewing the current time (Earth and Mars time zones), important announcements, and selected upcoming events. The tabbed panes contain more fullfeatured tools for navigating the space of files in the data repository, viewing event and staffing schedules, retrieving data products such as images, reading reports and summaries, and plotting telemetry data. The tools are interconnected and can call upon each other to provide functionality. For example, a report generated by the mission

staff at a particular meeting can be retrieved via the report-viewing tool or via the schedule-viewing tool. The user can then invoke the report viewer to display the report.

- *Middleware and web services.* The primary mechanism for sub-applications to obtain data is to invoke web services on a middleware server (Lea & Vinoski, 2003). Web services encode object structures and remote procedure invocations as XML, and use the normal HTTP protocols for transport. We use web services, in contrast to an earlier Java RMI architecture, as web services do not compromise firewall security by maintaining open connections. However, off-the-shelf XML/object translators don't present as richly recursive object model as RMI. The transition required modifying certain data structures to be simpler. The middleware is also responsible for vetting user identity and enforcing access control, and managing the movement of data to and from the repositories and CIP databases. Of particular relevance to the MER/CIP task is caching data: when an interesting data file arrives, 250 users may all want to see it simultaneously. The middleware server is driven by an Enterprise Java Bean model. It's notable in its use of both stateful and stateless session beans, and both container- and bean-managed entity beans.

- *Messaging.* Not all information flowing to the client comes as the result of client calls. One requirement is notifying clients when new information (e.g., changed schedules, new data files) appears. To handle this asynchronous notification, we have implemented a publish-and-subscribe messaging system, based on JMS and managed in the middleware. Clients subscribe to events of interest (like broadcast announcements and the appearance of new data files). Tools like the data acquisition monitor and broadcast announcer generate events that can match these subscriptions.

- *Data acquisition monitor.* The Rovers consume and produce a variety of data files, being presented with compilations of command sequences and returning the pictures and data from a variety of cameras and sensors. Scientists and operations staff want to know which commands have been or will be sent, and want to examine the data that has returned. This information resides on several different legacy MER Mission Data Servers. MER/CIP needs to know when things have arrived and where to find them. The data acquisition monitor process runs asynchronously, discovers new data files, inserts into the MER/CIP meta-database information about those files, and notifies the publish and subscribe system about them.

- *Meta-database.* MER/CIP (like the other systems in the InfoCore family) keeps two different kinds of data. Ordinary datasets (basically, data under control of MER/CIP, like user interface preferences) are kept in ordinary databases. Large data products (like Rover camera images) reside on the systems associated with the instruments that collected them. A meta-database keeps information about such files, including where to find them and searchable properties.. Dominating themes of the meta-database are (1) Scientific data is often naturally hierarchical. For example, a database may be composed of a series of experiments, each of which has a number of configurations. For each configuration, many identical steps may be performed (e.g., a series of photographs from a specific camera). The actual number of layers in this hierarchy varies among domains (and sometimes even varies within a domain) but the hierarchical nature is common. (2) The attributes of interest for any given experiment are numerous and vary from step to step. Thus, a conventional relational database organization of well-defined columns will prove inadequate. Instead, InfoCore systems use a relational database with mechanisms for expressing both "part-of" and dynamically defined relationships.
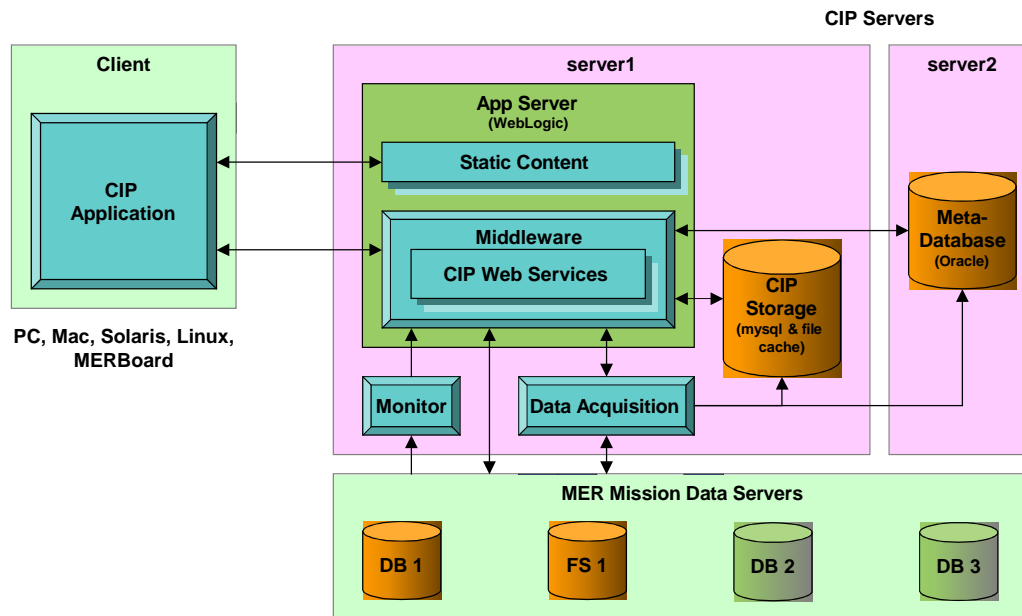
Figure 3: MER/CIP Software Architecture

4   Concluding Remarks

And astrology? Unlike most software projects, MER/CIP must be ready when the planets align. This hard deadline is proving to be an ambitious goal. At this writing, MER/CIP is nearing its version 1.1 release, consistently increasing its project management functionality and garnering enthusiasm from its prospective user base. In the complex mission operations environment, MER/CIP offers simple, easy-to-use tools that enable fast, secure access to critical information and afford MER/CIP users enhanced situational awareness. As the mission draws nearer, the need for an application that cuts across boundaries—between science and engineering data, Mars and Earth time zones, management and team member communications—to provide one-stop shopping for mission information is readily apparent. MER/CIP is rapidly moving beyond its initial role as mission enhancement and becoming an important contributor toward mission success.

References

Cornell/Athena Team (2003). Mission to Mars. http://athena.cornell.edu/the_mission/index.html

Jet Propulsion Laboratory (2003). 2003 Mars Exploration Rover Mission. http://mars.jpl.nasa.gov/missions/future/2003.html

Lea, D.; & Vinoski, S. (2003). Middleware for web services. *IEEE Internet Computing*, 7, 28–29/

Walton, J., Filman,R. E., Knight, C., Korsmeyer, D. J., & Lee, D. D. (2001). D3: A Collaborative Infrastructure for Aerospace Design. Workshop on Advanced Collaborative Environments, San Francisco, August 2001.

Walton, J., Filman,R. E., & Korsmeyer, D. J. (2000). The Evolution of the DARWIN System." *2000 ACM Symposium on Applied Computing* , Como, Italy, (pp. 971–977).